**EXPENSEMATE – AN EXPENSE TRACKER**

**A PROJECT REPORT**

*Submitted by*

CHAUHAN MALHAR JIGNESHKUMAR

(210130107035)

*In partial fulfillment for the award of the degree of*

**BACHELOR OF ENGINEERING**

***in***

**Computer Engineering**

**Government Engineering College, Sector-28, Gandhinagar**



**Gujarat Technological University, Ahmedabad**

**April,2025**



**Government Engineering College, Sector-28**

**Gandhinagar**

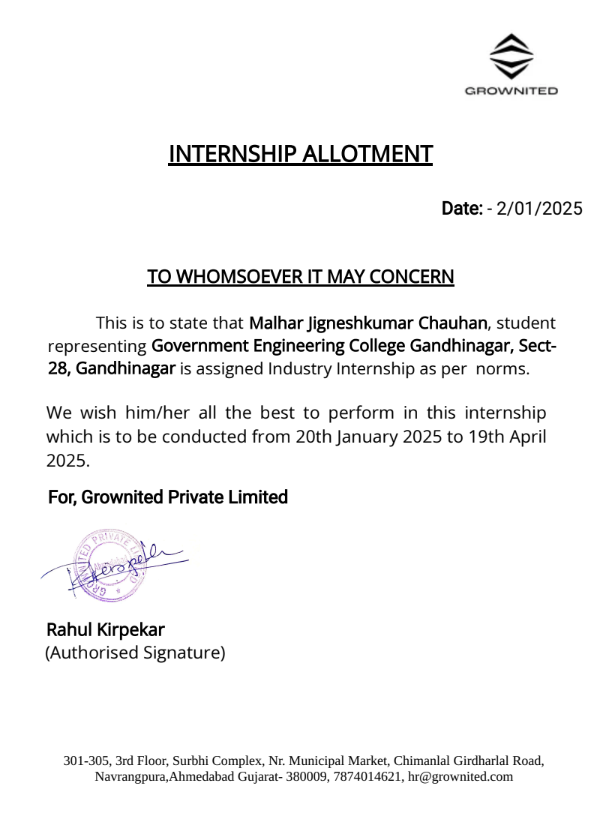
**CERTIFICATE**

This is to certify that the project report submitted along with the project entitled **ExpenseMate - An Expense Tracker** has been carried out by **Chauhan Malhar Jigneshkumar** under my partial fulfilment for the degree of Bachelor of Engineering in Computer Engineering, 8th Semester of Gujarat Technological University, and Ahmadabad during the academic year 2025-2026.

**Prof. Niranjan Prajapati Dr. D A Parikh**

**Internal Guide Head of Department**

**CONFIRMATION LETTER**

****



**Government Engineering College, Sector-28**

**Gandhinagar**

**DECLARATION**

We hereby declare that the Internship / Project report submitted along with the Internship / Project entitled **ExpenseMate - An Expense Tracker** submitted in partial fulfillment for the degree of Bachelor of Engineering in Computer Engineering to Gujarat Technological University, Ahmedabad, is a bonifide record of original project work carried out by me at **Grownited Private Limited** under the supervision of Prof. Niranjan Prajapati and that no part of this report has been directly copied from any students’ reports or taken from any other source, without providing due reference.

**Name of the Student Sign of Student**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

## **ACKNOWLEDGEMENT**

I am pleased to present this project report entitled **ExpenseMate – An Expense Tracker**. It is indeed a great pleasure and a moment of immense satisfaction for me to express my profound gratitude and indebtedness towards my guide **Prof. Niranjan Prajapati**, whose enthusiasm has been a constant source of inspiration for me.

I would also like to extend my sincere thanks to Dr. D. A. Parikh, Head of the Department, for their valuable guidance and for providing the necessary facilities to successfully complete this project report. My heartfelt thanks to the entire teaching and non-teaching staff for their support in data collection and software assistance, which played a crucial role in bringing this project to completion.

I am deeply grateful to my parents for their unwavering support and encouragement, which has been a prime source of motivation in completing this project without any obstacles.

Lastly, I would like to express my gratitude to my colleagues for their cooperation, useful suggestions, and all those who have directly or indirectly contributed to the successful completion of this project.

**Regards,**

**Chauhan Malhar Jigneshkumar (210130107035)**

**ABSTRACT**

An expense tracker is a tool or application that helps individuals or businesses monitor and manage their financial expenditures. The primary goal of an expense tracker is to provide a clear and comprehensive overview of spending patterns, enabling users to make informed financial decisions.

Typically, it involves logging income and expenses, categorizing transactions, setting budget limits, and generating reports. Advanced trackers may include features such as automated expense entry via bank integrations, visualizations like charts and graphs and alerts for overspending.

By offering insights into financial behavior, an expense tracker promotes better budgeting and financial planning.

**List of Figures**

Figure 5.2 Incremental Model 11

Figure 6.1 Expanse Tracker Architecture 13

Figure 7.1.1 Use Case Diagram Of User 16

Figure 7.1.2 Use Case Diagram Of Admin 17

Figure 7.2.1 Class Diagram 18

Figure 7.3.1 Sequence Diagram For User 19

Figure 7.3.2 Sequence Diagram For Admin 20

Figure 7.4.1 Activity Diagram 21

Figure 7.5.1 Level-0 Diagram 22

Figure 7.5.2 Level-1 Diagram 22

Figure 7.5.3 Level-2 Diagram 23

**List of Tables**

**Abbreviations**

|  |  |
| --- | --- |
| **Abbreviation** | **Full Form** |
| UX | User Experience |
| UI | User Interface |
| API | Application Programming Inteface |
| CRUD | Create, Read, Update, Delete |
| JSON | JavaScript Object Notation |
| OTP | One Time Password |
| FARM | FastAPI, React, MongoDB |
| FastAPI | Fast Asynchronous Python API |
| CORS | Cross-Origin Resource Sharing |
| SPA | Single-Page Application |
| REST | Representational State Transfer |

1. **OVERVIEW OF COMPANY**

**​**Grownited Private Limited is a startup cultivation firm based in Ahmedabad, Gujarat, dedicated to transforming innovative ideas into tangible realities. The company emphasizes the significance of youth entrepreneurship in shaping a modern and futuristic India.

* 1. **Services**
* Mobile App Development: Grownited brings concepts to life by developing Android mobile applications, combining passion and creativity with technical expertise.
* E-Commerce Development: The company provides e-commerce solutions for both B2B and B2C services, aiming to maximize the benefits of online selling for clients.
* Website Design: Understanding the importance of effective communication, Grownited's design team ensures that clients' messages are conveyed clearly and effectively through well-designed websites.
* Digital Marketing: Grownited's dedicated digital marketing team strives to deliver top-notch services, ensuring clients receive the best in the industry.
* Software Development: Combining design and technology, Grownited offers clients optimal UI/UX experiences and compatible websites and applications, recognizing the importance of a lasting first impression.

**2.2. Mission and Vision**

* Grownited serves as an open-source organization for individuals and groups who think differently, valuing ideas and striving to bring them to fruition. The company aims to become a hub of creativity and coherence by collaborating with idea cultivators, believing that nurturing the future is a shared responsibility. To support this vision, Grownited offers internships in over 15 categories for individuals aged 14–26, emphasizing the importance of experience in today's global scenario.
* By fostering a culture of innovation and providing comprehensive services, Grownited Private Limited plays a pivotal role in shaping the future of startups in modern India.

1. **INTRODUCTION TO EXPENSEMATE**
   1. **Overview**

**ExpenseMate** is a digital expense tracking solution designed to simplify financial management for individuals and organizations. It provides an intuitive platform to record, monitor, and analyze financial transactions efficiently. By offering features such as categorization, budgeting, and detailed reporting, ExpenseMate enhances financial awareness and helps users make informed financial decisions.

* 1. **Purpose and Importance**

Managing expenses manually can be tedious and error-prone. Traditional methods lack real-time insights, making it difficult to identify spending patterns and potential savings. ExpenseMate addresses these challenges by offering an automated and streamlined approach to financial tracking. The primary goal is to reduce manual effort, increase accuracy, and improve financial planning for users

* 1. **Key Features**
  + **Expense Recording:** Users can easily add, edit, and delete expenses with relevant details such as date, category, and amount.
  + **Categorization:** Expenses can be classified into predefined categories like food, transportation, rent, and entertainment, allowing for better financial organization.
  + **Budgeting:** Users can set monthly or category-based budgets to control their spending and avoid overspending.
  + **Visual Analytics:** Interactive charts and reports provide a clear understanding of spending habits and financial trends.
  + **Security and Privacy:** Data encryption and secure authentication ensure user privacy and data protection.
  1. **Scope of Project**

Managing and tracking daily expenses manually can be time-consuming and prone to errors. Users often struggle to maintain records, categorize expenses, and analyze spending patterns efficiently. With the **ExpenseMate**, users can digitally record and monitor their financial transactions with ease.

The scope of the project includes:

* **User Management:** Providing secure authentication and role-based access to users.
* **Expense Tracking:** Enabling users to log, categorize, and manage expenses efficiently.
* **Budget Management:** Allowing users to set budgets and receive alerts when exceeding limits.
* **Reporting and Analytics:** Generating insightful reports to help users analyze spending patterns.
* **Data Security:** Implementing encryption and security measures to protect user data.
  1. **Technology Review**

ExpenseMate leverages modern web technologies to provide a seamless and efficient experience.

* **FastAPI:** A high-performance Python-based framework, offering fast response times and easy API development.
* **React:** A component-based JavaScript library that provides a dynamic and interactive UI for users.
* **MongoDB:** A NoSQL database designed for scalability and flexibility in storing expense records.
* **RESTful API:** Ensures smooth communication between the frontend and backend services.
  1. **Literature Review**

Several studies highlight the importance of digital expense tracking in modern financial management:

* **Financial Planning Tools:** Research indicates that automated financial tracking solutions enhance budgeting and financial awareness by up to 40% compared to traditional methods.
* **User Engagement:** Studies show that mobile-friendly and cloud-based financial management tools improve user engagement and encourage proactive spending control.
* **Security in Financial Apps:** Literature emphasizes the role of encryption and secure authentication in maintaining user trust and data confidentiality.
* **Impact of Data Visualization:** Interactive dashboards and real-time analytics help users better understand their financial behaviors, leading to improved decision-making.
  1. **Conclusion**

**ExpenseMate** is a comprehensive and modern expense tracker designed to improve financial management. By leveraging automation, real-time data insights, and user-friendly features, it empowers users to take control of their finances effortlessly. Whether for personal use or business management, ExpenseMate serves as a reliable tool to ensure financial stability and growth.

1. **SOFTWARE & HARDWARE REQUIREMENTS**
   1. **Functional Requirements**

Functional requirements outline the core functionalities of the Expense Tracker website, ensuring seamless data processing, user interactions, and system operations. These requirements define how the system should handle transactions, manage expenses, generate reports, and authenticate users.

Additionally, these functionalities are supported by non-functional requirements, which establish performance, security, and usability constraints to enhance system efficiency and user experience.

* + 1. **Hardware Requirements**
* Intel Core i5 processor or higher
* 8GB RAM or more
* 512GB SSD storage
* Windows 11 operating system
* Stable internet connection for hosting and testing
  + 1. **Software Requirements**
* Visual Studio Code / PyCharm (for coding and development)
* Postman (for API testing)
* Command Prompt / Terminal (for executing commands and scripts)
* Web Browser (for application testing and debugging)
  1. **Non-Functional Requirements**

Non-functional requirements define the quality attributes and constraints of the system, ensuring efficiency, security, and scalability. Unlike functional requirements, which specify what the system does, non-functional requirements determine how well the system performs.

Non-functional requirements include:

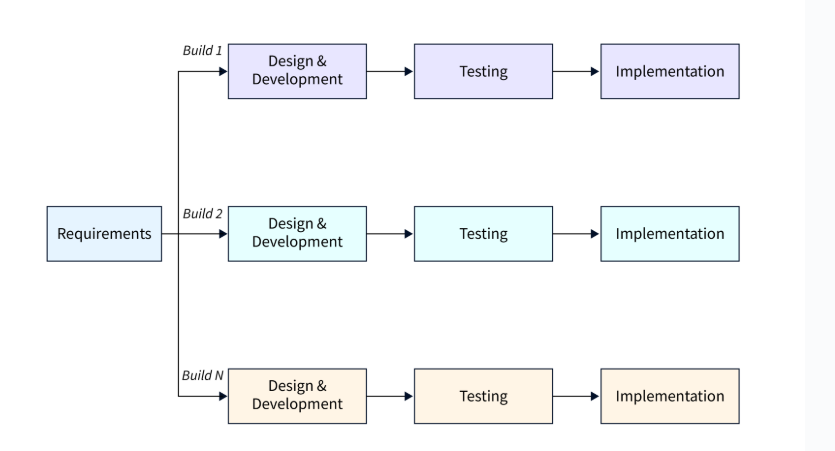
* **Reliability –** The system must operate consistently without failures.
* **Scalability –** The system should efficiently handle increased user loads.
* **Performance** – Fast response times and optimized resource utilization are essential.
* **Security –** Robust measures must be in place to prevent unauthorized access and data breaches.
* **Usability –** The interface should be intuitive and user-friendly.
* **Maintainability –** The system should be easy to update, modify, and enhance as needed**.**

1. **PROCESS MODEL**
   1. **Incremental Process Model**

The **Incremental Process Model** is a combination of one or more **Waterfall Models**. In this model, the project requirements are divided into multiple **modules**, and each module is developed separately. Once developed, these modules are integrated with other modules to form the complete system.

During the development of each module, the **Waterfall Model** is followed individually. Each developed module is a **standalone feature** and can be delivered to end users for use. Additional modules are integrated incrementally as new features in subsequent releases.

One of the key advantages of the Incremental Model is that **there is no need to wait for all modules to be developed and integrated**. Since each module is independent and has no dependencies on others, the project can be delivered in phases. The process continues until all requirements are met, and the entire system is fully developed.

****

[ Figure 4.1 Incremental Model ]

* 1. **Advantages**
* Enables early delivery of working software within the development lifecycle.
* Provides flexibility, making it less costly to modify scope and requirements.
* Simplifies testing and debugging through smaller iterations.
* Allows customers to provide feedback after each build.
* Reduces initial delivery costs.
* Enhances risk management by identifying and addressing high-risk components in iterations.
  1. **Disadvantages**
* Requires good planning and design.
* Demands a well-defined system structure before dividing it into increments.
* Typically incurs higher overall costs compared to the Waterfall model.
  1. **When to Use This Model?**
* Core requirements are established, but some details may evolve over time.
* Early market release is a priority.
* The project involves new or emerging technologies.
* Skilled resources are not readily available.
* High-risk features and objectives must be addressed.
* Suitable when system requirements are well-defined and clearly understood.

1. **SYSTEM ANALYSIS**
   1. **Feasibility Analysis**

The primary goal of the feasibility study is to assess whether developing the **Expense Tracker** system is financially and technically viable. This study involves analysing the core problem, gathering relevant information, and determining the necessary inputs, processing methods, and expected outputs while considering system constraints.

A preliminary investigation evaluates whether the system will effectively assist users in managing their financial transactions.

The feasibility study consists of three key aspects, each playing a crucial role in determining the success of the system:

* + 1. **Technical Feasibility:**

The **Expense Tracker (ExpenseMate)** system is technically feasible as we have the required technology for its implementation. The system is developed using **React.js** for the frontend and **FastAPI** for the backend.

For development, we used **VS Code** and **PyCharm** as primary tools. The backend is powered by **FastAPI**, ensuring fast and efficient API communication. **MongoDB** is used as the database to store all user expenses and transaction records. Authentication and session management are handled securely using token-based authentication.

By leveraging these technologies, the system ensures seamless functionality, data security, and high performance.

* + 1. **Economic Feasibility:**

The **Expense Tracker(ExpenseMate)** system is economically feasible as the cost of development is minimal, making it a good investment for users

The system is a **one-time investment** that does not require expensive hardware or software. It is developed using **open-source technologies** like **React.js, FastAPI, and MongoDB**, reducing licensing costs.

Since users can access the system through a **web application**, there is **no need for special tools or devices**. The system improves efficiency by automating expense tracking, **reducing manual effort** and **saving time**, making it a cost-effective solution.

* + 1. **Schedule Feasibility:**

Schedule feasibility ensures that the project can be completed within the given time frame. The **Expense Tracker** project is planned for **3 months**, divided into the following phases:

* **Month 1** – Requirement gathering, feasibility study, and system design.
* **Month 2** – Development and implementation.
* **Month 3** – Testing, debugging, and deployment.

The timeline is **realistic and achievable**, ensuring efficient completion within the given period.

* + 1. **Operational Feasibility:**

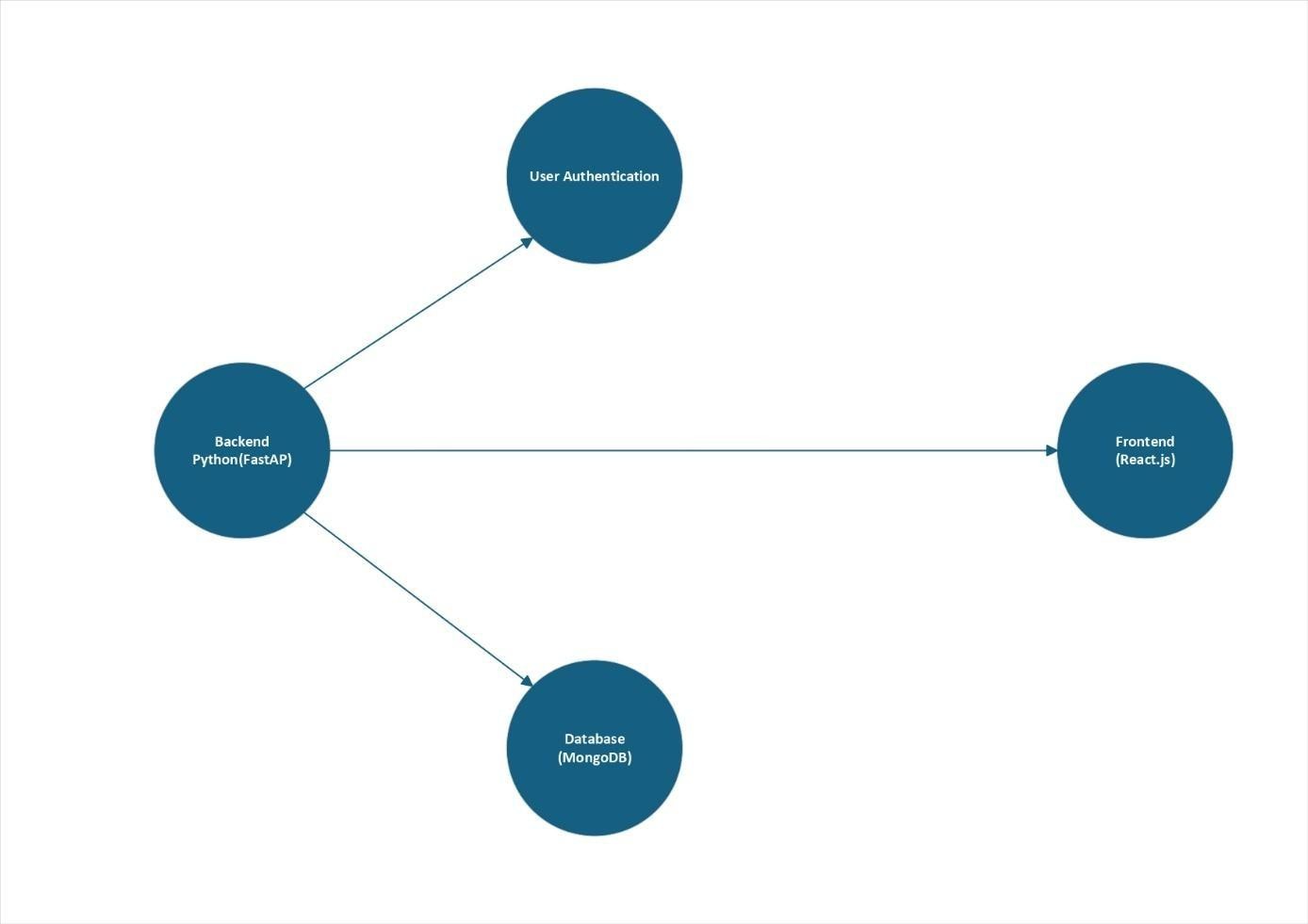
The **Expense Tracker** system is designed to be **user-friendly and efficient**. It allows users to **add, edit, and delete expenses**, track financial records, and categorize expenses easily. The **Admin** has full control over user management, reports, and system settings.

* **No extra training** is required to use the system.
* The system ensures **easy operation with enhanced features**.
* Provides **quick access to financial information** for better expense tracking and management.
  + 1. **Implementation Feasibility:**

The organization provides all necessary resources and software for developing the Expense Tracker website. The required tools and technologies are readily available, making implementation smooth. Additional software and libraries were downloaded from the internet as needed, ensuring feasibility in development.

1. **PROJECT PLANNING**
   1. **About ExpenseMate**

**ExpenseMate** is an intuitive and efficient expense tracking application designed to help users manage their personal and business finances with ease. Developed using **FastAPI, React.js, and MongoDB,** it allows users to **record transactions, categorize expenses, set budgets**, and visualize financial insights through interactive dashboards.



[ Figure 6.1 ExpenseMate Architecture ]

With FastAPI for backend, React.js for frontend, and MongoDB for storage, this system ensures fast, secure, and efficient expense tracking.

* 1. **Features of ExpenseMate**
* **Secure Access:** Only registered users can access the system.
* **Expense & Income Management:** Users can add, edit, and delete their income and expense records.
* **Category Customization:** Users can create and manage categories for expenses and income.
* **Budget Tracking:** Users can set monthly budgets and monitor their spending against them.
* **Visual Analytics:** A dashboard with visual reports helps users analyze financial data.
* **Data Privacy & Security:** Secure authentication mechanisms ensure user data is protected.
* **Expense Report Exporting:** Users can export reports for better financial planning.
  1. **Need of an ExpenseMate**

In today's fast-paced digital world, managing personal and business finances effectively is crucial. Many individuals struggle with tracking their income, expenses, and savings, leading to financial mismanagement.

The Expense Tracker Application provides an intelligent, automated solution for recording, categorizing, and analyzing expenses in real-time. Built using FastAPI (backend), React.js (frontend), and MongoDB (database), the application enables users to track financial transactions seamlessly and maintain control over their budget.

**Key Benefits:**

* **Time-saving:** Automates expense tracking, reducing manual effort.
* **Reduced financial stress:** Helps users maintain financial discipline.
* **Better decision-making:** Provides clear insights into spending habits.
  1. **Mechanism of ExpenseMate**

Users can log into the application and record their income and expenses in different categories. The system automatically analyzes spending patterns and provides insights through visual charts and reports.

**Core Functionalities:**

* **User Registration & Login:** Secure authentication for user access.
* **Transaction Management:** Users can add, edit, and delete transactions.
* **Category-wise Tracking:** Expenses are categorized for better management.
* **Dashboard & Reports:** Users can view spending trends through interactive charts.
* **Admin Control:** The admin can manage user accounts and monitor the system.

By leveraging **FastAPI, React.js, and MongoDB**, the system ensures fast, secure, and efficient expense tracking.

* 1. **Transaction List**

The Transaction List displays all financial transactions recorded by users. Each transaction includes:

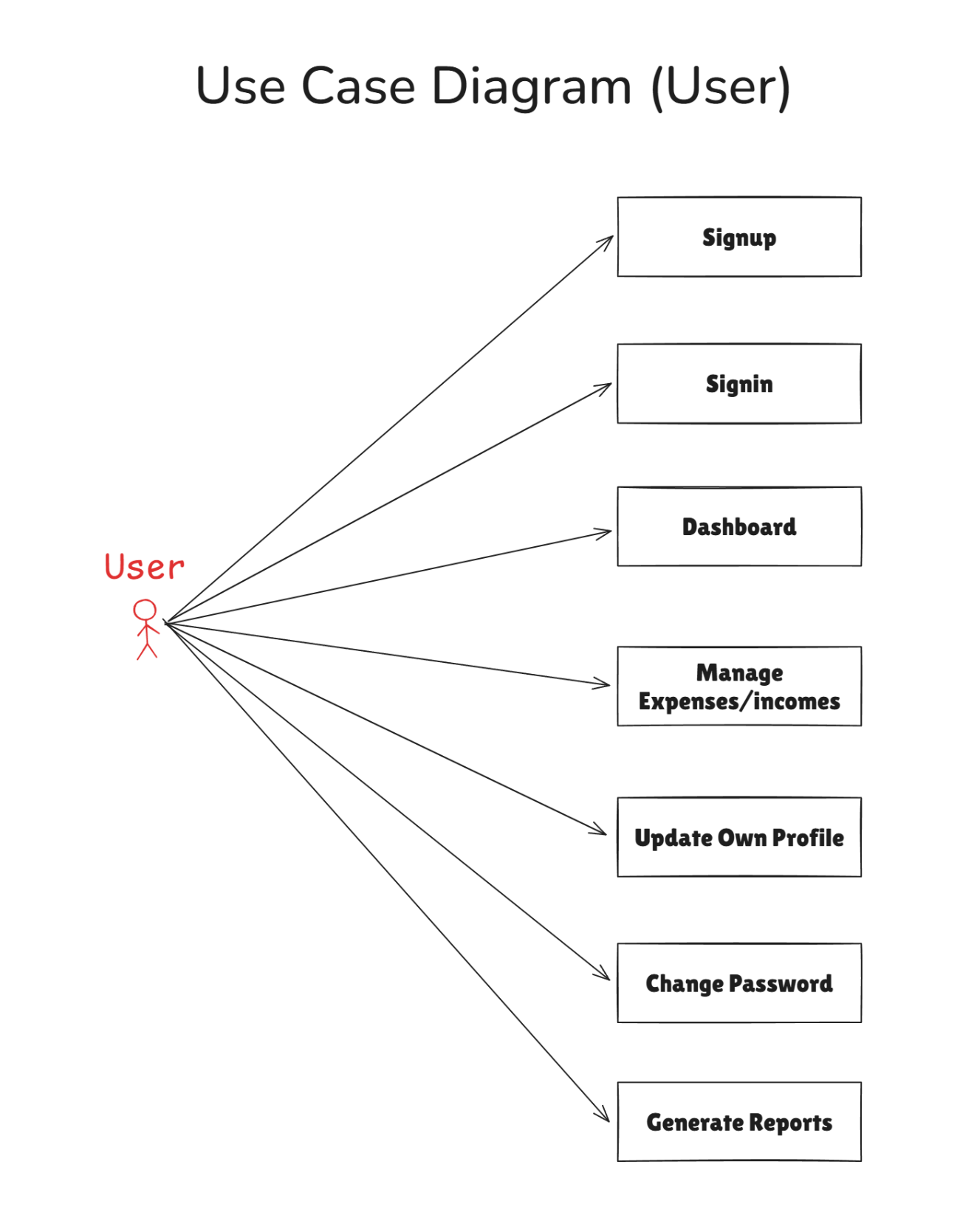
* Transaction Type : Expense or Income
* Amount
* Category
* Date
  1. **Add, Edit and Delete Expenses**
* Users can **add** expenses with multiple details, including amount, category, and date.
* They can **edit** or **delete** expenses when necessary.
* The **admin** has the ability to manage all expenses added by users.
  1. **Feedback System:**

Through the feedback system, users can contact the Admin. When the Admin receives a feedback message, they can respond directly to the users. This enables smooth communication between users and the Admin. Additionally, users can provide reviews based on their experience with expenses or financial management features. Admin can view and analyze the feedback given by users for continuous improvement.

* 1. **Contact us**

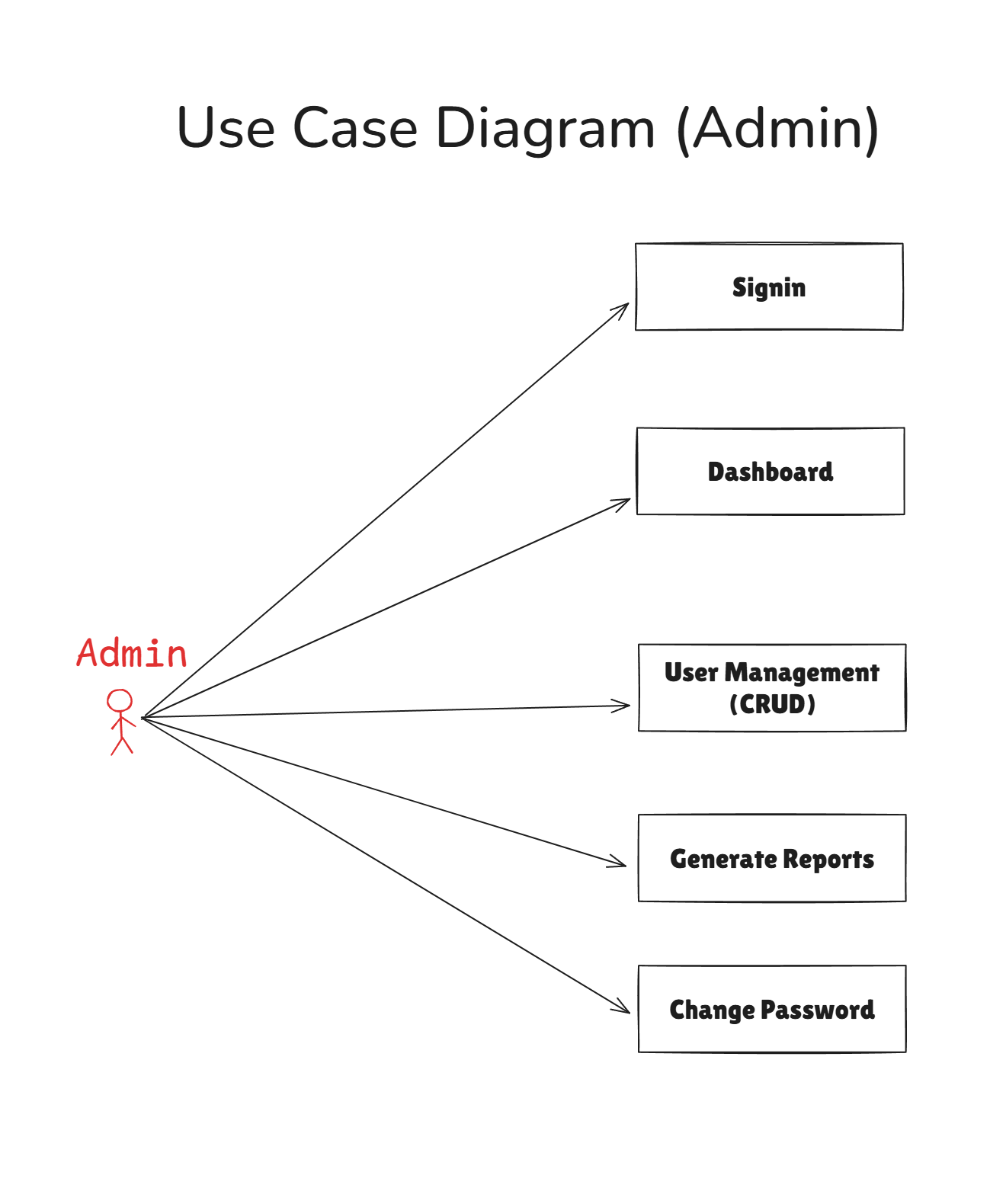
Through this page, users can contact the admin by providing suggestions or reporting any issues related to the application. This ensures smooth communication and helps in improving the user experience.

1. **SYSTEM DESIGN**
   1. **Use Case Diagrams**
      1. **Use Case Diagram of User**



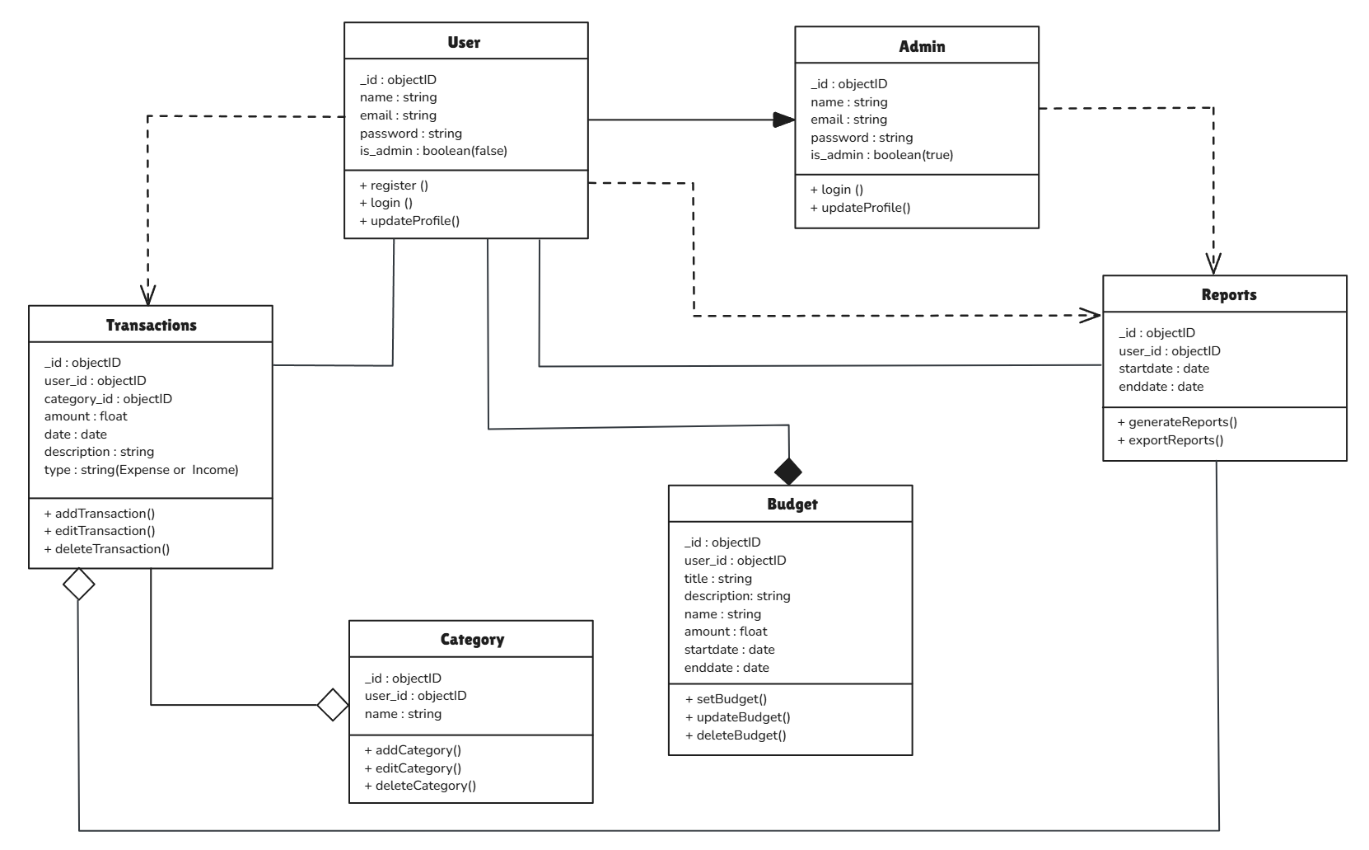
[ Figure 7.1.1 Use Case Diagram of User]

* + 1. **Use Case Diagram of Admin**



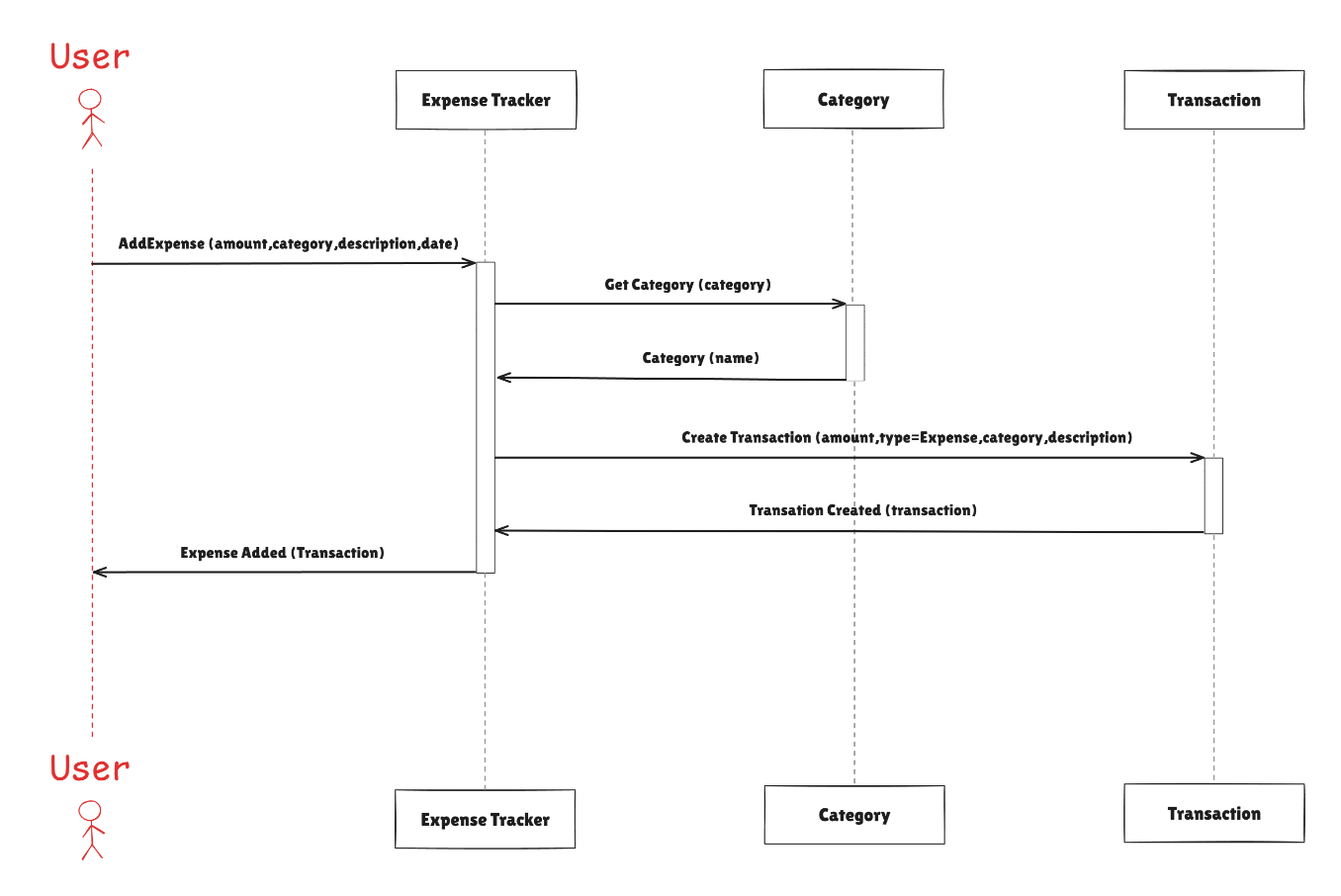
[ Figure 7.1.2 Use Case Diagram of Admin]

* 1. **CLASS DIAGRAM**



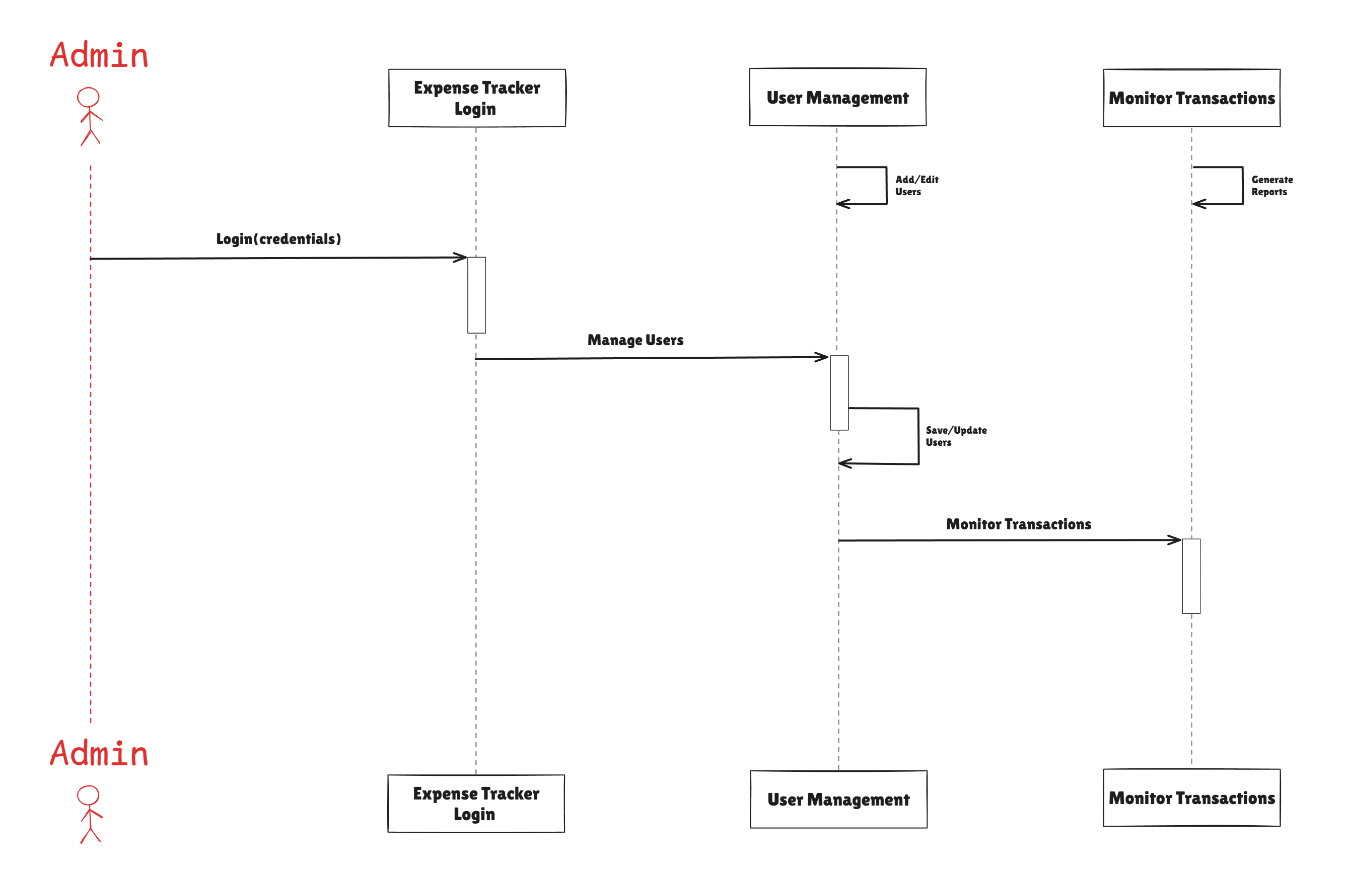
[ Figure 7.2.1 Class Diagram (ExpenseMate) ]

* 1. **SEQUENCE DIAGRAM**
     1. **Sequence Diagram of User**



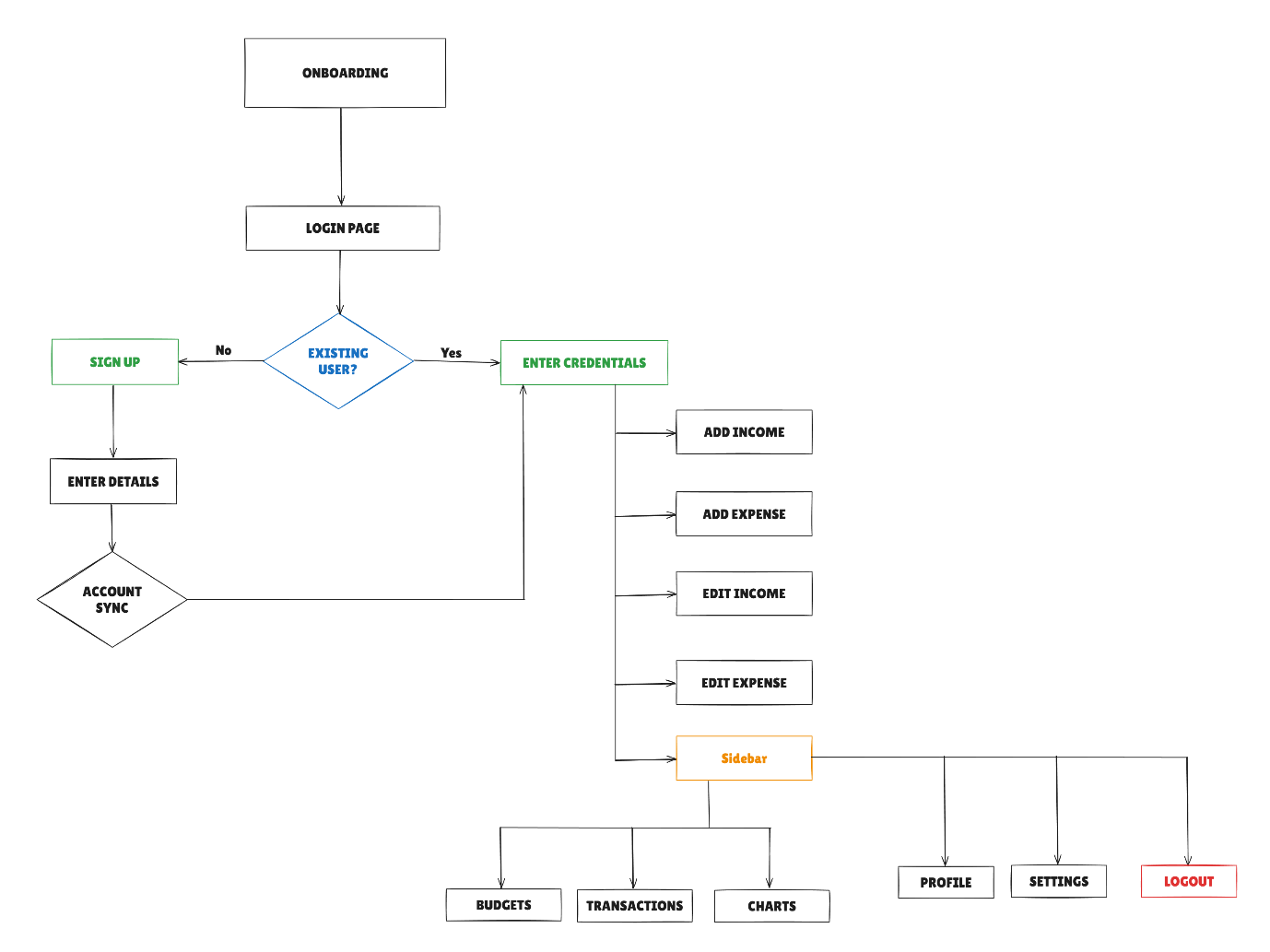
[ Figure 7.3.1 Sequence Diagram of User]

* + 1. **Sequence Diagram of Admin**



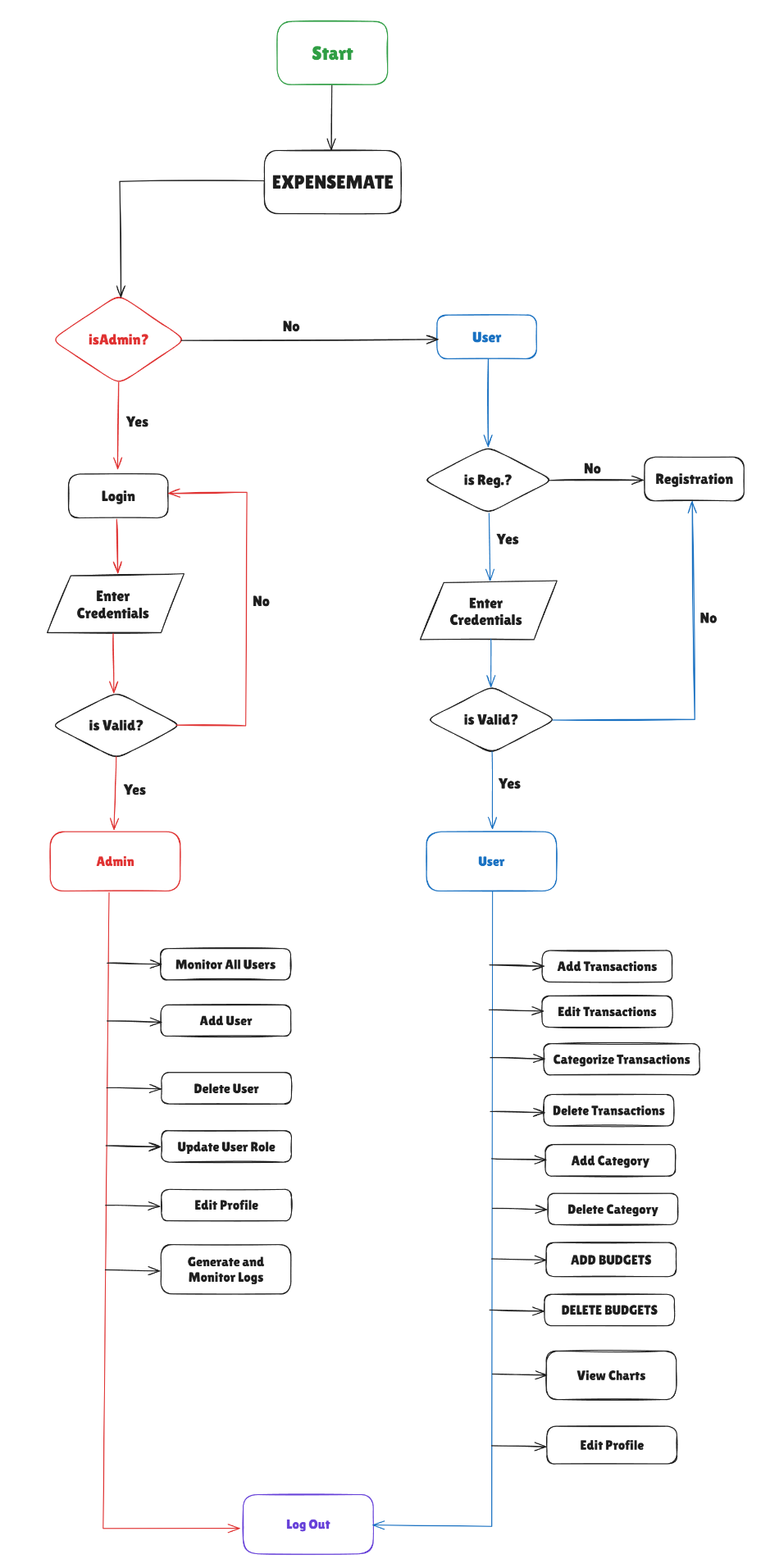
[ Figure 7.3.2 Sequence Diagram of Admin]

* 1. **ACTIVITY DIAGRAM**



[ Figure 7.4.1 Activity Diagram ]

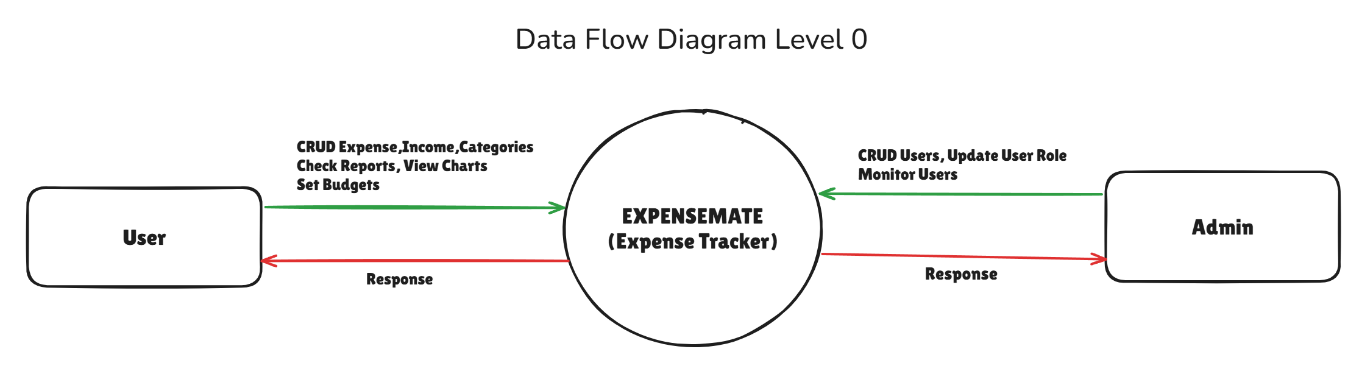
* 1. **Flow Chart**



[ Figure 7.5.1 Flow Chart ]

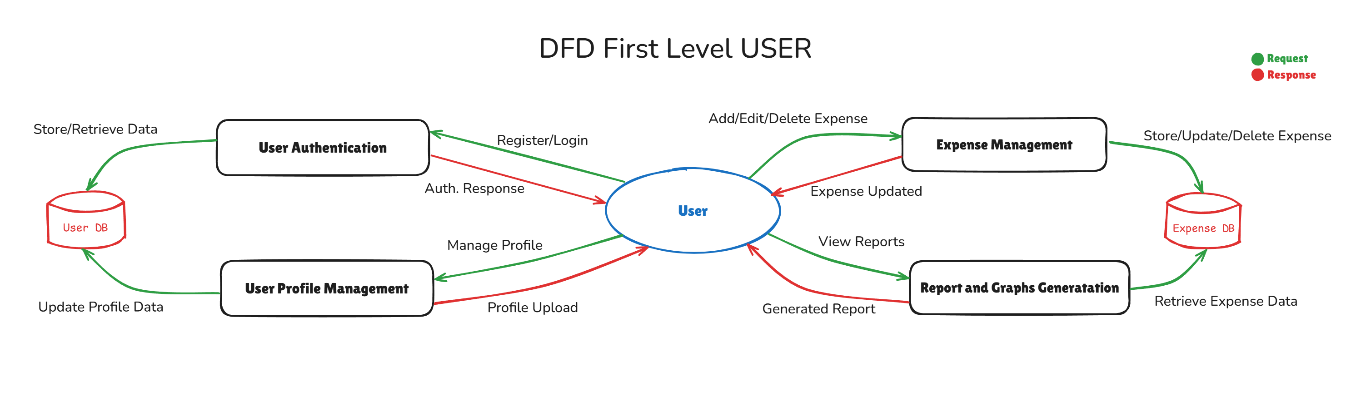
* 1. **Data Flow Diagrams**

**7.6.1 Data Flow Diagram level 0**



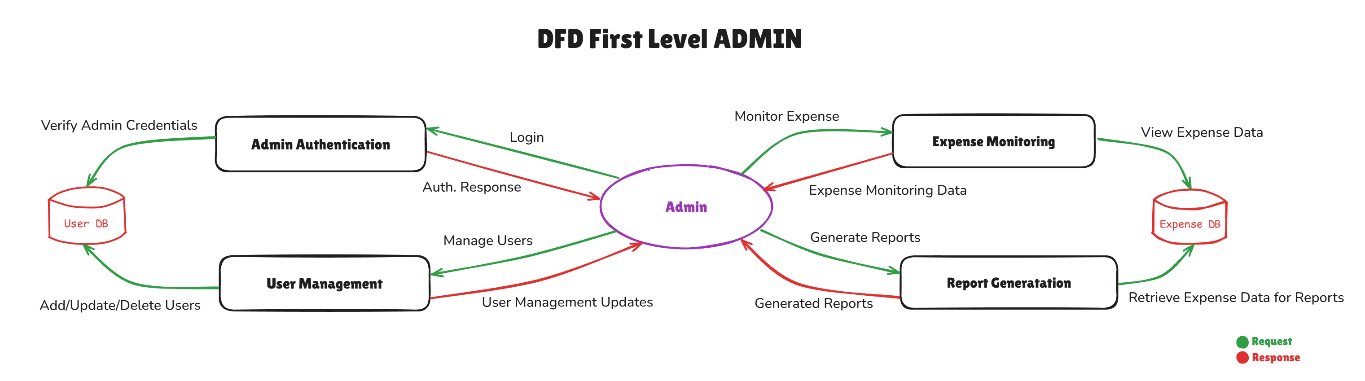
[ Figure 7.6.1 DFD Level 0]

**7.6.2 User DFD Level 1**

****

[ Figure 7.6.2 User DFD Level 1]

**7.6.2 Admin DFD Level 1**

****

[ Figure 7.6.2 Admin DFD Level 1]